Fake News Detection and Generalizability

Marie MEYER
Student at ENSAE PARIS
marie.meyer@ensae.fr

Abstract

This project aims to implement an NLP pipeline for fake news detection and test its generalization capabilities. We compare the performance of five types of features —Bag of Words, TF-IDF, Word2Vec, BERT, and a set of linguistic cues— on fake news classification. Using the ISOT dataset, we train multiple models and find that most combinations achieve excellent results, with F1 scores above 0.99 in over half of the cases. We then select the top-performing model for each feature set and evaluate them on a different dataset (Fake or Real News). Results drop significantly, revealing that models trained on one dataset do not generalize well to another. This underlines the challenge of fake news detection in real-world conditions.

1 Description of the problem

The detection of fake news is a well-known NLP problem, which has become more and more important in recent years as the spread of misinformation has accelerated through social media and other digital platforms. With the increasing influence of online media on public opinion, the need for reliable methods to detect false information is more important than ever. The task of fake news detection can be seen as a supervised classification task in NLP, where the goal is to distinguish between real and fake articles based on their textual content. This involves analyzing diverse linguistic features and textual representations to train machine learning models, and to evaluate which combinations of features and models are more efficient (e.g. accuracy-wise) to classify fake news.

2 State of the art

Early approaches to fake news detection relied on traditional machine learning classifiers such as Logistic Regression, Passive Aggressive Classifier, SVMs and Random Forest applied to simple features like bag-of-words counts, TF-IDF vectors, N-grams or lexical features such as average word length, length of article or number of adjectives, as per Sharma et al. [9] and Khanam et. al.'s articles [6]. These models provided a baseline by capturing lexical and stylistic differences between fake and real news articles.

The introduction of distributed word embeddings like Word2Vec and GloVe allowed models to incorporate semantic relationships between words [7]. These representations improved performances but were limited by their context-independence.

Recent advances in transformer-based models like BERT, have significantly improved fake news detection by allowing to capture semantic and long-distance dependencies in sentence. Fine-tuning such pre-trained models on labeled fake news datasets has consistently yielded state-of-the-art performance (Kaliyar et al. [5]).

The article we focused on for our project addresses the Generalisability of Fake News Detection Models [4], and focuses on testing whether Fake News detection models are able to generalize well on datasets they were not trained on. This idea comes from the observation that the state of the

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

art Fake News detection models mentioned above obtain extremely good results, with accuracies almost always close to 99%. But do these models retain their performance when applied to different databases ? The article shows that most often, they do not.
The authors trained six models (AdaBoost, Gradient Boosting, Logistic Regression, Random Forest, MLP and SVM) on five different features (Bag of Words counts, TF-IDF counts, Word2Vec embeddings, Bert embeddings and linguistic cues) extracted from four separate datasets (ISOT, Fake or Real, Kaggle Fake News Competition and FakeNewsNet). They then performed cross-evaluation, in which each model trained on a given dataset is tested on the three others.
Comparing results between the tests on the same vs. on different datasets showed large drops in accuracy when cross-testing (from 37% to 53% accuracy drops depending on the dataset), when same dataset testing achieved accuracies of 99%. This failure to generalize effectively exposes a critical limitation in current fake news detection systems, and emphasizes the need for cross-domain evaluation.

3 Experiment proposition and justification

After exploring and preprocessing the ISOT dataset, it quickly became clear that fully replicating the results of the original article was too ambitious. Indeed, training 6 models on 5 different features across 4 datasets would have required computational and time resources we did not have. Therefore, we decided to focus my project on the ISOT dataset [1], which was developped by the Information Security and Object Technology lab at the University of Victoria, and primarily contains political news articles from the US and around the world. It is split into 2 CSV files : Fake.csv and True.csv, and includes 23,481 fake articles collected from unreliable websites, and 21,417 real articles published on Reuters.com. Each entry contains the article title, text, subject, and date.

Following an initial data analysis (see 4), we applied the preprocessing and feature extraction steps described in Hoy et al.'s paper (see sections 5 and 6) and trained four models on each feature. The models chosen were Random Forest, AdaBoost, Gradient Boosting and Logistic Regression. MLP and SVM were left out due to resource constraints and because their results in the original study were very close to the other models.

These models were then evaluated on the ISOT dataset using stratified cross-validation. Finally, to tie this project to the original article, we also tested the trained models on a second dataset: the Fake or Real dataset from Kaggle[2]. This dataset contains 3164 fake articles and 3171 real articles, each entry containing its title, its text and its label. We applied the same preprocessing and feature extraction steps and then tested the ISOT-trained models on this new dataset to assess their generalizability.

4 Data analysis

4.1 ISOT dataset :

The dataset contains 44898 observations and 5 columns (article title, text, subject, date and label). 52% of the articles are labeled as fake and 48% as real. As shown in Figure 1, the majority of articles focus on political news, both from the US and around the world.

[1] `https://www.kaggle.com/datasets/csmalarkodi/isot-fake-news-dataset`

[2] `https://www.kaggle.com/datasets/jillanisofttech/fake-or-real-news/data`

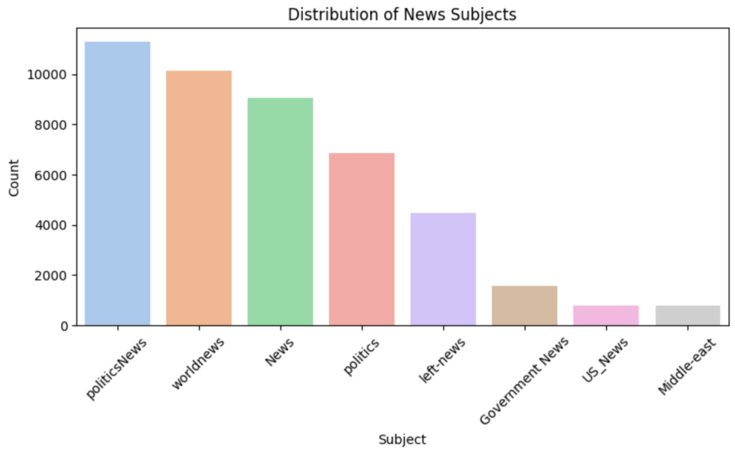


Figure 1: Subject distribution in the ISOT dataset

Figure 2 compares the distributions of text and title lengths between fake and real articles. Overall, we observe that text lengths exhibit a similar distribution across both classes, although fake articles tend to have more high extreme values, indicating unusually long texts. Title length distributions appear to be more distinct between fake and real articles : fake articles generally have longer titles and show more variance in their length compared to real articles.

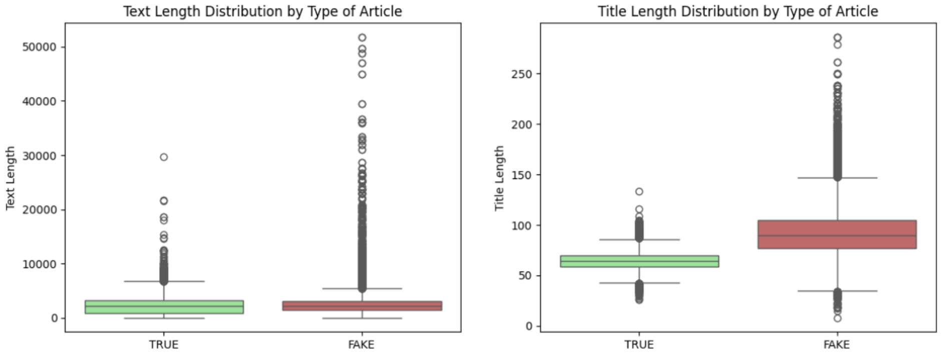


Figure 2: Boxplots of the text and title lengths distributions for fake and real articles

Finally, to get a general idea of the topics discussed in the articles, we perform light topic exploration through word clouds. We compute the wordclouds separately for fake and real articles using both word counts and TF-IDF Vectorization. The results appear to be consistent across both vectorizers used. We observe that majority of the articles mention US political figures with words such as Trump, president, U.S., Obama and Clinton, suggesting a strong focus on american politics. The TF-IDF Vectorizer allows us to detect a few additional themes compared to the word counts, with words like stock and fund appearing, hinting to financial topics. However, we cannot differentiate between the word clouds of fake and real articles, indicating that the same topics are being treated similarly.

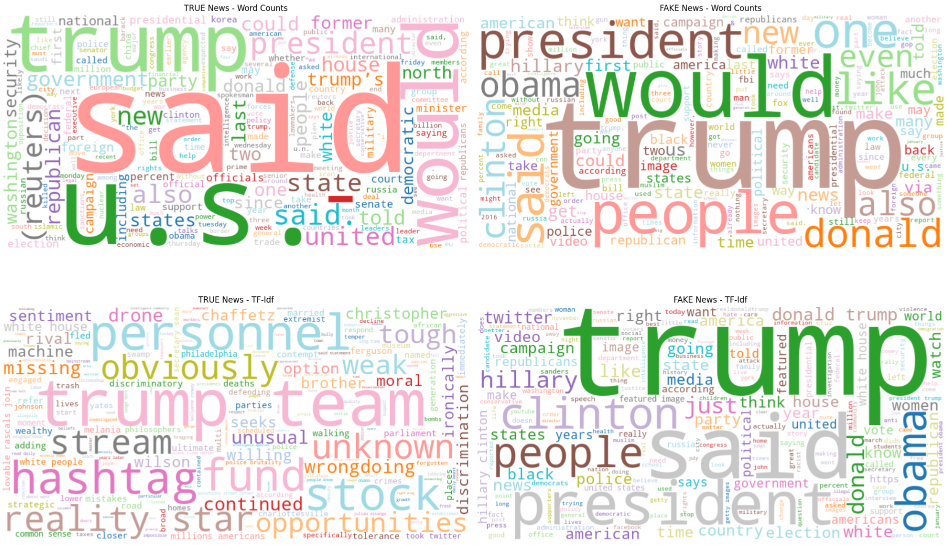


Figure 3: Word clouds for the ISOT dataset

4.2 Fake or Real dataset :

We conducted the same analyses on the Fake or Real dataset, which is considerably smaller than the ISOT one, with only 6334 observations. As shown in Figure 4, we observe similar patterns when analyzing the texts' and titles' lengths distributions : fake articles tend to have longer texts and titles than real ones, with slightly greater variance in length as well.

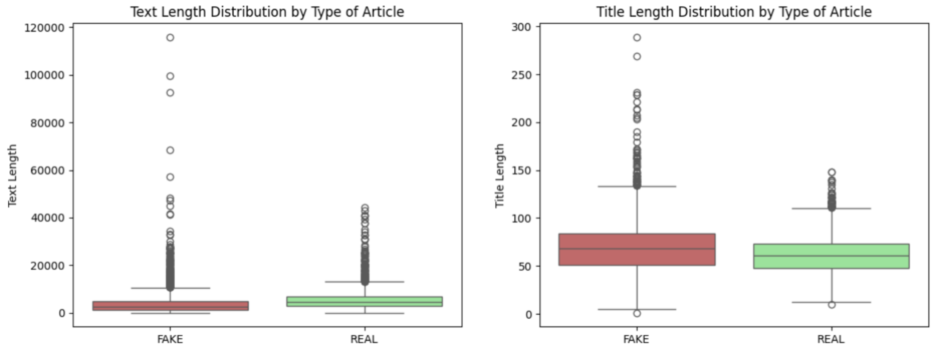


Figure 4: Boxplots of the text and title lengths distributions for fake and real articles

The topic modeling for this dataset (cf Figure 5) leads to the same conclusions as previously : the articles mostly focus on american political figures, with words like Trump, Clinton and Reagan appearing most frequently. As before, it is difficult to notice any difference between the word clouds of fake and real articles. Similarly, switching from the use of TD-IDF to simple word counts does not produce significant changes.

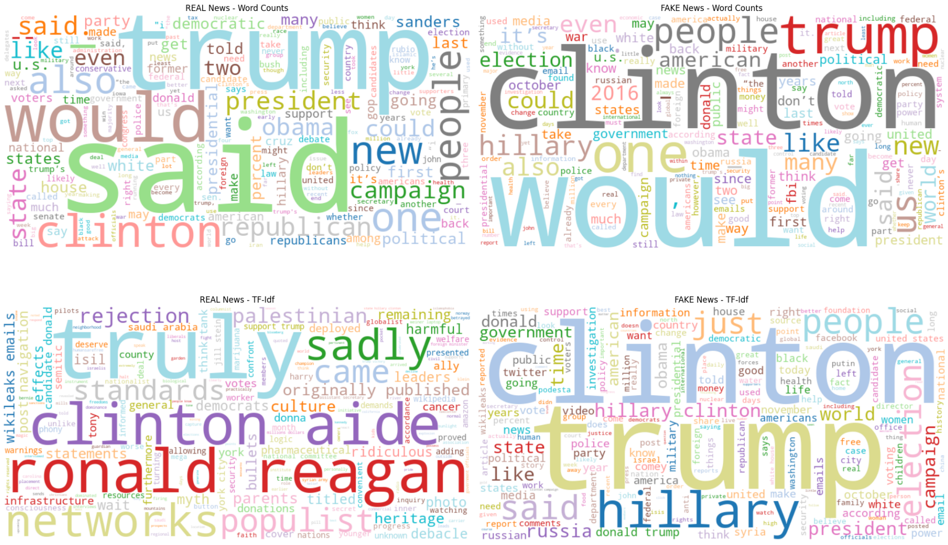


Figure 5: Word clouds for the Fake or Real dataset

5 Data preprocessing

The data preprocessing steps are as described in the original paper from Hoy et al. ; we first convert the article texts to lowercase and apply tokenization. We then remove stopwords and apply lemmatization to reduce the words to their root forms. The goal of these steps is to minimize unwanted noise from the texts.
For models like Bert and Word2Vec, which rely on contextual information to compute embeddings, we need to keep the full sentences and only convert the text to lowercase. Finally, linguistic cues feature extraction (see 6) requires the originial text format, as it involves computing statistical properties directly from the raw text.

We apply the same preprocessing steps across both datasets.

6 Feature extraction

As in Hoy et al.'s article, we derive 5 features from each dataset :

1. Bag of Words counts - a frequency-based approach that reflects how many times each word appears in the corpus.
2. TF-IDF vectorizer counts - another frequency-based approach that reflects the importance of a word in a given article, relative to its frequency across the entire corpus.
3. Word2Vec embeddings - each word is represented as a vector in a large vector space. To aggregate word-level embeddings into document-level embeddings, article vectors are computed as the mean of the word vectors in the article. We used a pretrained Word2Vec model for this step.
4. BERT embeddings - unlike Word2Vec, BERT generates context-dependent embeddings by incorporationg both the position and the semantic meaning of the words. We used the pretrained BERT uncased model for this step.
5. Linguistic cues - features that describe how text is written, such as average sentence length, punctuation use, or the frequency of pronouns and verbs. We extracted 32 of them, described in the table in Annex A.1.

7 Model experiments

To evaluate fake news detection performance, we trained four supervised classifiers on the ISOT dataset: AdaBoost, Gradient Boosting, Logistic Regression and Random Forest. For each model, we tested the five feature representations described earlier. All experiments were conducted using stratified 5-fold cross-validation to ensure balanced splits of real and fake articles across folds, as per Hoy et al. Model performance was assessed using accuracy, precision, recall, and F1 score, with the 'FAKE' class treated as the positive label, since we want to detect fake news.

To assess how well these models generalize beyond the ISOT dataset, we performed a cross-dataset evaluation on the Fake or Real dataset. First, for each feature type, we selected the best-performing model out of the four, based on F1 score. These five models were retrained on the full ISOT dataset and saved for later inference. We then evaluated each model on the unseen Fake or Real dataset. For each model–feature combination, we computed the same classification metrics as before.

8 Result analysis

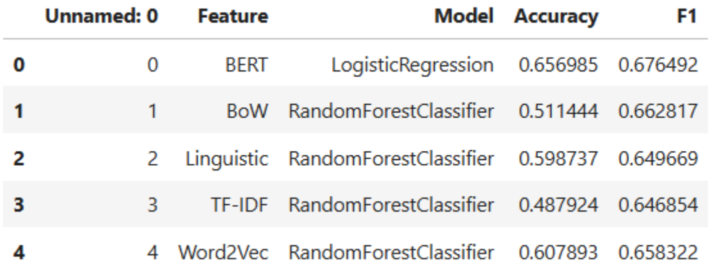
8.1 Intra-dataset evaluation

The results, ordered by decreasing F1 score, can be found in Annex 7. Overall, Random Forest, Gradient Boosting and Logistic Regression models performed best, especially when combined with TF-IDF or Bag of Words features. The top-performing combination (Random Forest with TF-IDF) achieved an F1 score of 0.9976.
Feature-wise, Word2Vec and linguistic cues performed slightly less well in comparison, with lower F1 scores, especially for Logistic Regression and AdaBoost.

However, the gap between the best and worst performing combinations is very small, with only a 0.08 difference in F1 score. The lowest score is of 0.9142 (linguistic cues with Logistic Regression) and still reflects strong classification performance.

8.2 Cross-dataset evaluation

Figure 6 shows the cross-dataset evaluation results, which shows how well the models trained on the ISOT dataset perform on the Fake or Real dataset. The table includes accuracy and F1 score for each feature–model combination.



	Unnamed: 0	Feature	Model	Accuracy	F1
0	0	BERT	LogisticRegression	0.656985	0.676492
1	1	BoW	RandomForestClassifier	0.511444	0.662817
2	2	Linguistic	RandomForestClassifier	0.598737	0.649669
3	3	TF-IDF	RandomForestClassifier	0.487924	0.646854
4	4	Word2Vec	RandomForestClassifier	0.607893	0.658322

Figure 6: Cross-evaluation results

For every feature, we observe a significant drop in accuracy and F1 score on the Fake or Real dataset. While they were all greater than 90% on the ISOT Dataset, here F1 scores only range from 0.65 to 0.68 with the TF-IDF accuracy even going as low as 48%.
This shows that the models do not generalize well and likely overfit to ISOT-specific patterns.

Conclusion

We explored five different feature types extracted from the ISOT Dataset to create a fake news detection algorithm. After extensive data preprocessing and feature extraction, we trained four models (Random Forest, AdaBoost, Gradient Boosting and Logistic Regression) on each feature, resulting in a total of 20 models. The results obtained were excellent, with all models achieving accuracies and F1 scores above 91%, and nearly half exceeding 99%.

To test their capacity to generalize to different datasets, we processed and extracted the same features on a second political news dataset, and applied the best-performing model for each feature to the new data. The resuts showed significant drop in performance, with accuracies and F1 scores decreasing by around 30%. This highlights poor generalizability across datasets.

This study suggests that while traditional features and models can fit well on a single dataset, building robust fake news detectors, adapted to multiple domains, requires more diverse training data and possibly more sophisticated, domain-adaptive approaches.

References

[1] Hadeer Ahmed, Issa Traore, and Sherif Saad. "Detecting opinion spams and fake news using text classification". In: *Security and Privacy* 1 (Dec. 2017), e9. DOI: 10.1002/spy2.9.

[2] Aaron Carl T. Fernandez and Madhavi Devaraj. "Computing the Linguistic-Based Cues of Fake News in the Philippines Towards its Detection". In: *Proceedings of the 9th International Conference on Web Intelligence, Mining and Semantics*. WIMS2019. Seoul, Republic of Korea: Association for Computing Machinery, 2019. ISBN: 9781450361903. DOI: 10.1145/3326467.3326490. URL: https://doi.org/10.1145/3326467.3326490.

[3] Nathaniel Hoy and Theodora Koulouri. *A Systematic Review on the Detection of Fake News Articles*. 2021. arXiv: 2110.11240 [cs.CL]. URL: https://arxiv.org/abs/2110.11240.

[4] Nathaniel Hoy and Theodora Koulouri. "Exploring the Generalisability of Fake News Detection Models". In: *2022 IEEE International Conference on Big Data (Big Data)*. 2022, pp. 5731–5740. DOI: 10.1109/BigData55660.2022.10020583.

[5] Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. "FakeBERT: Fake news detection in social media with a BERT-based deep learning approach". In: *Multimedia tools and applications* 80.8 (2021), pp. 11765–11788.

[6] Zeba Khanam et al. "Fake news detection using machine learning approaches". In: *IOP conference series: materials science and engineering*. Vol. 1099. 1. IOP Publishing. 2021, p. 012040.

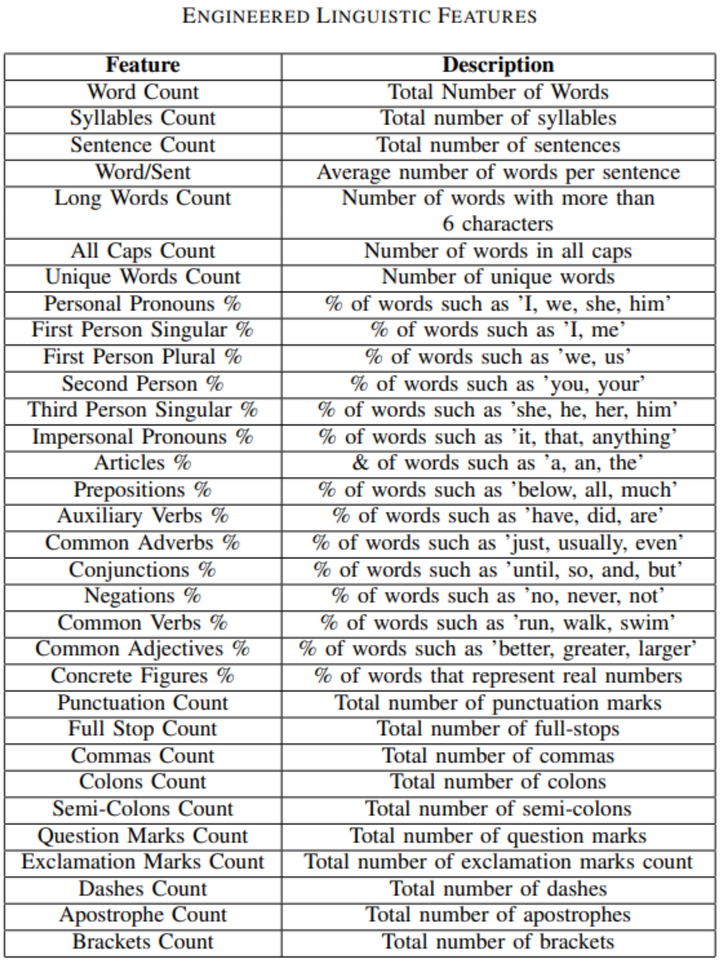
[7] Marwa Naili, Anja Habacha Chaibi, and Henda Hajjami Ben Ghezala. "Comparative study of word embedding methods in topic segmentation". In: *Procedia Computer Science* 112 (2017). Knowledge-Based and Intelligent Information Engineering Systems: Proceedings of the 21st International Conference, KES-20176-8 September 2017, Marseille, France, pp. 340–349. ISSN: 1877-0509. DOI: https://doi.org/10.1016/j.procs.2017.08.009. URL: https://www.sciencedirect.com/science/article/pii/S1877050917313480.

[8] Karishnu Poddar, Geraldine Bessie Amali D., and K.S. Umadevi. "Comparison of Various Machine Learning Models for Accurate Detection of Fake News". In: *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*. Vol. 1. 2019, pp. 1–5. DOI: 10.1109/i-PACT44901.2019.8960044.

[9] Uma Sharma, Sidarth Saran, and Shankar M Patil. "Fake news detection using machine learning algorithms". In: *International Journal of creative research thoughts (IJCRT)* 8.6 (2020), pp. 509–518.

A Annex

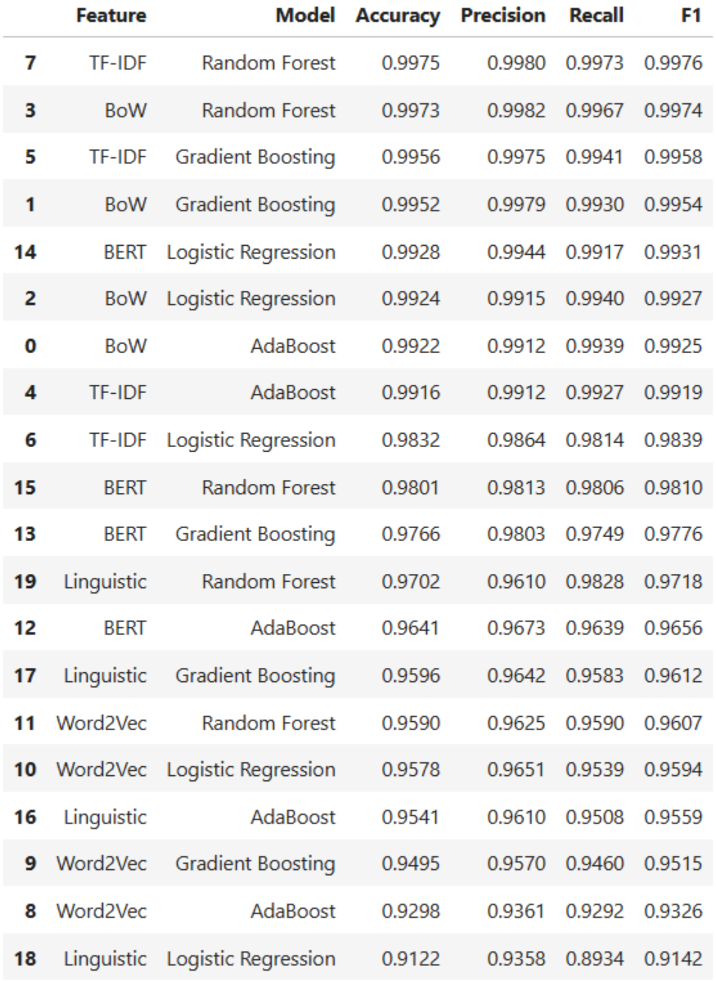
A.1 Linguistic cues description



ENGINEERED LINGUISTIC FEATURES

Feature	Description
Word Count	Total Number of Words
Syllables Count	Total number of syllables
Sentence Count	Total number of sentences
Word/Sent	Average number of words per sentence
Long Words Count	Number of words with more than 6 characters
All Caps Count	Number of words in all caps
Unique Words Count	Number of unique words
Personal Pronouns %	% of words such as ’I, we, she, him’
First Person Singular %	% of words such as ’I, me’
First Person Plural %	% of words such as ’we, us’
Second Person %	% of words such as ’you, your’
Third Person Singular %	% of words such as ’she, he, her, him’
Impersonal Pronouns %	% of words such as ’it, that, anything’
Articles %	& of words such as ’a, an, the’
Prepositions %	% of words such as ’below, all, much’
Auxiliary Verbs %	% of words such as ’have, did, are’
Common Adverbs %	% of words such as ’just, usually, even’
Conjunctions %	% of words such as ’until, so, and, but’
Negations %	% of words such as ’no, never, not’
Common Verbs %	% of words such as ’run, walk, swim’
Common Adjectives %	% of words such as ’better, greater, larger’
Concrete Figures %	% of words that represent real numbers
Punctuation Count	Total number of punctuation marks
Full Stop Count	Total number of full-stops
Commas Count	Total number of commas
Colons Count	Total number of colons
Semi-Colons Count	Total number of semi-colons
Question Marks Count	Total number of question marks
Exclamation Marks Count	Total number of exclamation marks count
Dashes Count	Total number of dashes
Apostrophe Count	Total number of apostrophes
Brackets Count	Total number of brackets

A.2 Results



	Feature	Model	Accuracy	Precision	Recall	F1
7	TF-IDF	Random Forest	0.9975	0.9980	0.9973	0.9976
3	BoW	Random Forest	0.9973	0.9982	0.9967	0.9974
5	TF-IDF	Gradient Boosting	0.9956	0.9975	0.9941	0.9958
1	BoW	Gradient Boosting	0.9952	0.9979	0.9930	0.9954
14	BERT	Logistic Regression	0.9928	0.9944	0.9917	0.9931
2	BoW	Logistic Regression	0.9924	0.9915	0.9940	0.9927
0	BoW	AdaBoost	0.9922	0.9912	0.9939	0.9925
4	TF-IDF	AdaBoost	0.9916	0.9912	0.9927	0.9919
6	TF-IDF	Logistic Regression	0.9832	0.9864	0.9814	0.9839
15	BERT	Random Forest	0.9801	0.9813	0.9806	0.9810
13	BERT	Gradient Boosting	0.9766	0.9803	0.9749	0.9776
19	Linguistic	Random Forest	0.9702	0.9610	0.9828	0.9718
12	BERT	AdaBoost	0.9641	0.9673	0.9639	0.9656
17	Linguistic	Gradient Boosting	0.9596	0.9642	0.9583	0.9612
11	Word2Vec	Random Forest	0.9590	0.9625	0.9590	0.9607
10	Word2Vec	Logistic Regression	0.9578	0.9651	0.9539	0.9594
16	Linguistic	AdaBoost	0.9541	0.9610	0.9508	0.9559
9	Word2Vec	Gradient Boosting	0.9495	0.9570	0.9460	0.9515
8	Word2Vec	AdaBoost	0.9298	0.9361	0.9292	0.9326
18	Linguistic	Logistic Regression	0.9122	0.9358	0.8934	0.9142

Figure 7: Feature-model combinations performances

A.3 Cross-evaluation performances

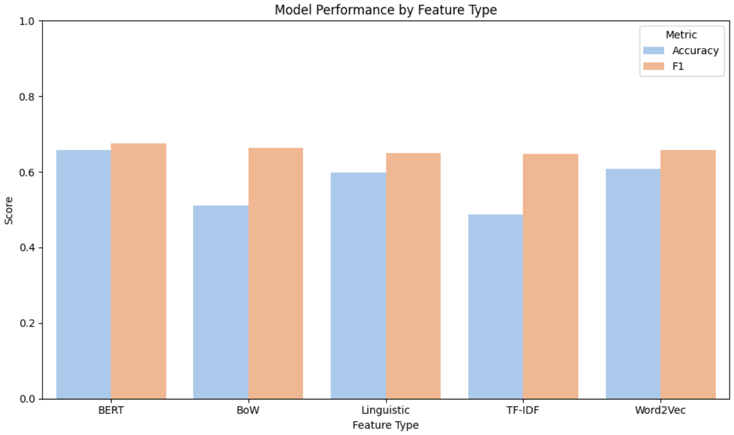


Figure 8: Cross-evaluation model performances by feature type